An Approach to Using Non Safety-Assured Programmable Components in Modest Integrity Systems

Peter Bishop[1,2], Kostas Tourlas[2], and Nick Chozos[2]

[1] Centre for Software Reliability, City University, London
Northampton Square, London, EC1V 0HB, United Kingdom
[2] Adelard LLP
Northampton Square, London, EC1V 0HB, United Kingdom
{pgb,kt,nc}@adelard.com

Abstract. Programmable components (like personal computers or smart devices) can offer considerable benefits in terms of usability and functionality in a safety-related system. However there is a problem in justifying the use of programmable components if the components have not been safety justified to an appropriate integrity (e.g. to SIL 1 of IEC 61508). This paper outlines an approach (called LowSIL) developed in the UK CINIF nuclear industry research programme to justify the use of non safety-assured programmable components in modest integrity systems. This is a seven step approach that can be applied to new systems from an early design stage, or retrospectively to existing systems. The stages comprise: system characterisation, component suitability assessment, failure analysis, failure mitigation, identification of additional defences, identification of safety evidence requirements, and collation and evaluation of evidence. In the case of personal computers, there is supporting guidance on usage constraints, claim limits on reliability, and advice on "locking down" the component to maximise reliability. The approach is demonstrated for an example system. The approach has been applied successfully to a range of safety-related systems used in the nuclear industry.

Keywords: Programmable components, safety integrity, safety assurance.

1 Introduction

Programmable components like personal computers (PCs) or smart devices can offer considerable benefits in terms of usability and functionality in a safety-related system. However there is a problem in justifying the use of programmable components if they have not been safety justified to an appropriate integrity (e.g. to SIL 1 of IEC 61508 [3]).

To address this issue, the UK Control and Instrumentation Nuclear Industries Forum (CINIF) sponsored a research project (called LowSIL) to assure the safety of modest integrity systems that used "non safety-assured programmable components" (NSPC). The development of the guidance took place in a series of projects:

- A review of approaches actually used when PCs were used in safety-related systems.
- Production of guidance for using PCs in safety-related systems.

E. Schoitsch (Ed.): SAFECOMP 2010, LNCS 6351, pp. 377–390, 2010

- Generalisation of the approach to other types of NSPC.
- Updating the guidance in response to user feedback after the guidance had been applied to actual systems.

In parallel to development of the guidance, we also undertook research, primarily on Microsoft Windows-based PCs, to establish:

- Options for "locking down" Windows to make it more reliable and secure.
- Experimental validation by applying the guidance on a test PC
- Monitoring over extended periods to establish realistic reliability figures for typical applications running under Windows in a locked-down state.

The LowSIL guidance has been applied within the nuclear industry and has been updated to reflect user feedback In the sections below we present the most recent version of the guidance that has been produced. This guidance is intended for use when:

- Failure of the system can affect nuclear safety, environmental protection or industrial safety, the integrity of plant actuations, safety-related information presented to operators, or the safety integrity of components or calibration data that will be used in the plant at some time in the future.
- The required integrity of the system safety function is at or below SIL 1. Typically no more than 10^{-1} failures per demand or 10^{-4} dangerous failures per hour
- The system contains one or more NSPC. An NSPC is a programmable device such as a PC, a programmable logic controller, or a configurable device such as a smart sensor, and does not have sufficient assurance of its safety integrity.

The guidance can be applied equally to the assessment of new and pre-existing systems. Examples of systems where the guidance has been applied are PC-based monitoring and logging control systems, maintenance support and control of equipment tests.

2 Safety Assurance Context

In order to assure the safety of the modest integrity system we first need to identify the context in which the system operates as illustrated in Fig.1 below.

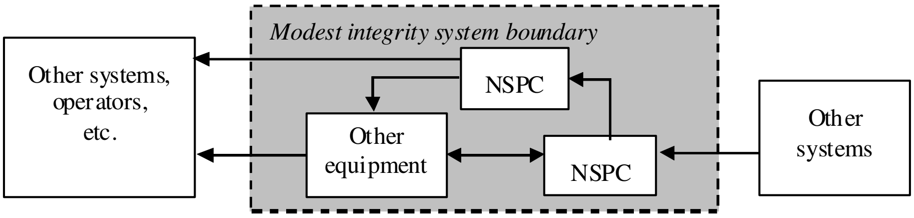
Modest integrity system boundary
Other systems, operators, etc.
NSPC
Other equipment
NSPC
Other systems

Fig. 1. Safety Assurance Context

In this context we have:

- The modest integrity system, containing one or more NSPC (possibly of different types). Other parts of the modest integrity system can be affected by NSPC failures, but could also contain defences to detect and mitigate failures.
- External systems beyond the modest integrity system boundary (like other C&I equipment, system operators, etc). Failures that propagate beyond the system boundary could affect these external systems. But again there could be defences in the wider system that could mitigate the failures.

3 Structure of the Guidance

The guidance contains the following elements:

- Generic guidance that can be applied to any system containing NSPCs.
- Annexes containing guidance about specific NSPCs.

This structure was chosen because it could be readily extended to include new types of NSPC. Currently the focus has been on Windows-based PCs, but the structure is designed to be readily extended to other NSPC, like smart devices or PLCs.

The specific guidance contains:

- Limitations on use, e.g. Windows PCs are precluded from use for real-time control.
- Guide performance figures, such as reliability, performance, fail-safety and diagnostic coverage.
- Lock-down guidance to enhance reliability and security. Better guide figures for reliability can be used if the component is locked-down.

4 Overview of the Assessment Process

For a new modest integrity system, or replacement system, this assessment process should start as early in the lifecycle as possible, while it is still feasible to determine and influence the modest integrity system design and implementation.

The steps in the LowSIL process and the resultant documentation outputs are shown in Fig. 2. The decision points are shown as diamonds and represent points where the modest integrity system could be rejected as unsuitable.

It can be seen that the process consists of seven discrete steps, which are summarised below:

Step 1 Characterise the plant context, modest integrity system, embedded NSPC(s) and their types (PC, PLC, etc). Include a clear statement of whether the modest integrity system is a new system, a replacement system or a pre-existing system.

Step 2 Characterise the requirements placed on each NSPC within the system and assess if the component can feasibly meet them. This characterisation considers:

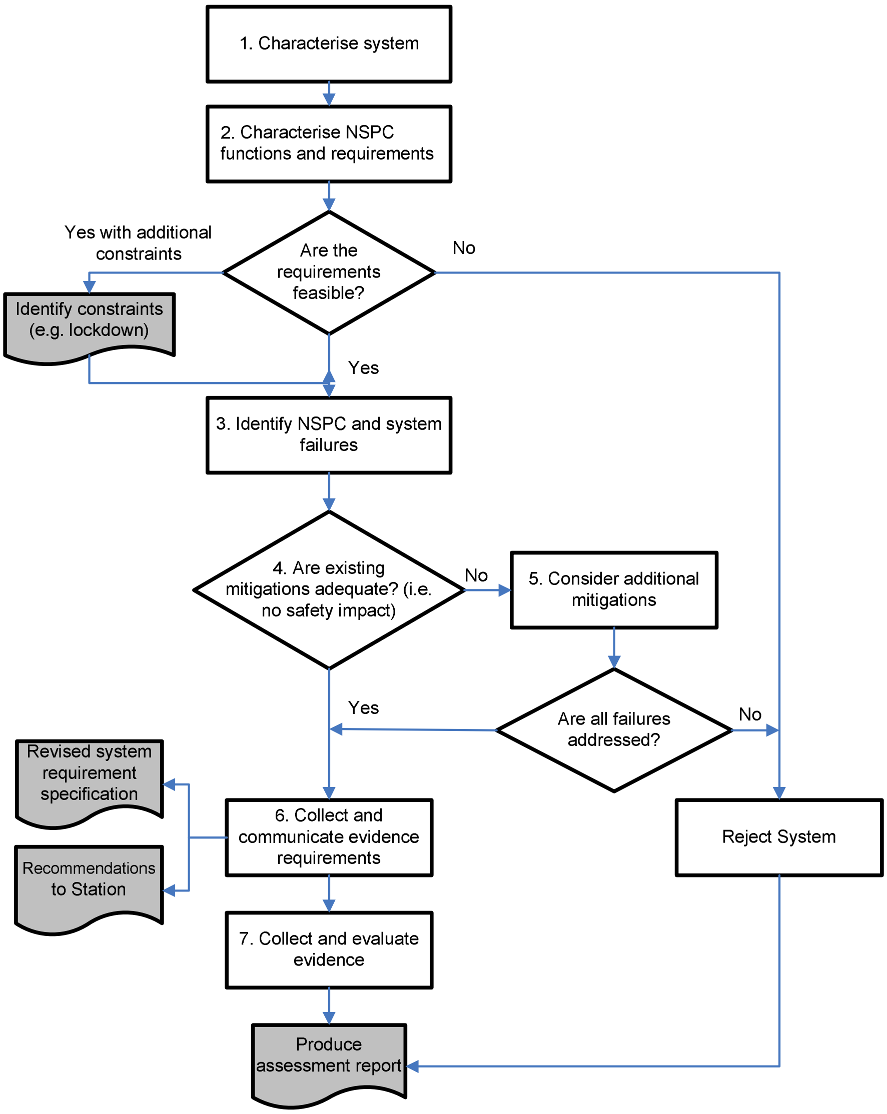
1. Characterise system
2. Characterise NSPC functions and requirements
Are the requirements feasible?
Yes with additional constraints
No
Identify constraints (e.g. lockdown)
Yes
3. Identify NSPC and system failures
4. Are existing mitigations adequate? (i.e. no safety impact)
No
5. Consider additional mitigations
Yes
Are all failures addressed?
No
Revised system requirement specification
Recommendations to Station
6. Collect and communicate evidence requirements
Reject System
7. Collect and evaluate evidence
Produce assessment report

Fig. 2. LowSIL Safety Assessment Process

- The type of function (e.g., advisory, monitoring, control) performed by each NSPC within the system.
- The performance targets for the function, especially reliability, time-response, etc.

In a prospective assessment (of a new/replacement system), compare the performance targets with reasonable limits provided in the relevant NSPC Annex. If the targets exceed these limits, appropriate changes to the design are required (e.g. “hardening” a PC operating system by a lock-down of its features). These need to be included in the requirements specification for the system. If this is not possible, the system is not considered acceptable.

Similarly, in a retrospective assessment (of a pre-existing system), continued use of the existing system is not acceptable if the performance requirements exceed the guide limits unless there is a statistically valid basis from prior use showing that the system can meet its targets.

Step 3 For each NSPC within the system, identify how it may fail, including performance failures such as slow response. Consider how a failure by the NSPC could lead to a hazard at the system’s boundary. For example, an incorrect command or instruction issued by the system could affect the safety performance of other systems or the wider plant.

Step 4 Assess the safety impact of hazardous NSPC failures, by considering the effectiveness of the mitigation available for these failures.

If there are residual NSPC failures that are not effectively mitigated or controlled, the existing mitigation should be strengthened or new mitigation added. If this is not possible, the system is not considered acceptable.

Step 5 If shown to be necessary by Step 4, consider additional mitigation options and select those that are feasible for the system at hand.

In retrospective assessments this additional mitigation is likely to be procedural. Make specific recommendations about the changes and additions to the existing procedures that will be needed in order to justify the continued use of the system.

In prospective assessments, precedence should be given to technical mitigations. Those that are feasible will result in design change proposals and new system requirements for inclusion in the requirements specification for the system (and possibly in the supply contract).

If at the end of this step there are still failures of the NSPCs that cannot be shown to be effectively mitigated or controlled by the additional measures, the system cannot be assessed as acceptable.

Step 6 Collect the requirements that emerged from the previous steps (new design requirements, changes to procedures, etc). A successful assessment must show that these new requirements have been met, so in this step, determine the additional documentation, verification and validation demonstration and other activities (such as independent assessment of the software) that will be needed as evidence. Ensure that these evidence requirements are included in the Supplier’s contract, communicated to the end-user, etc., as appropriate.

Step 7 Collect and evaluate the evidence available. Produce the assessment report, resulting in a clear recommendation as to whether the system can be accepted or not.

5 Example Application of the Guidance

To illustrate the approach, the assessment steps will be applied to a modest integrity system that forms part of a Unit Maintenance Facility (UMF) in a power plant. The units maintained by the UMF will be used later in plant operation and hence the system is safety-related.

5.1 Step 1: Characterisation of the Modest Integrity System and Environment

The modest integrity system incorporates a PC server, a terminal, a printer and local area network. It is part of the UMF and acts as the Human-Machine Interface (HMI) for the facility's Safety Logic System (SLS). We will refer to it as the "HMI" system for brevity.

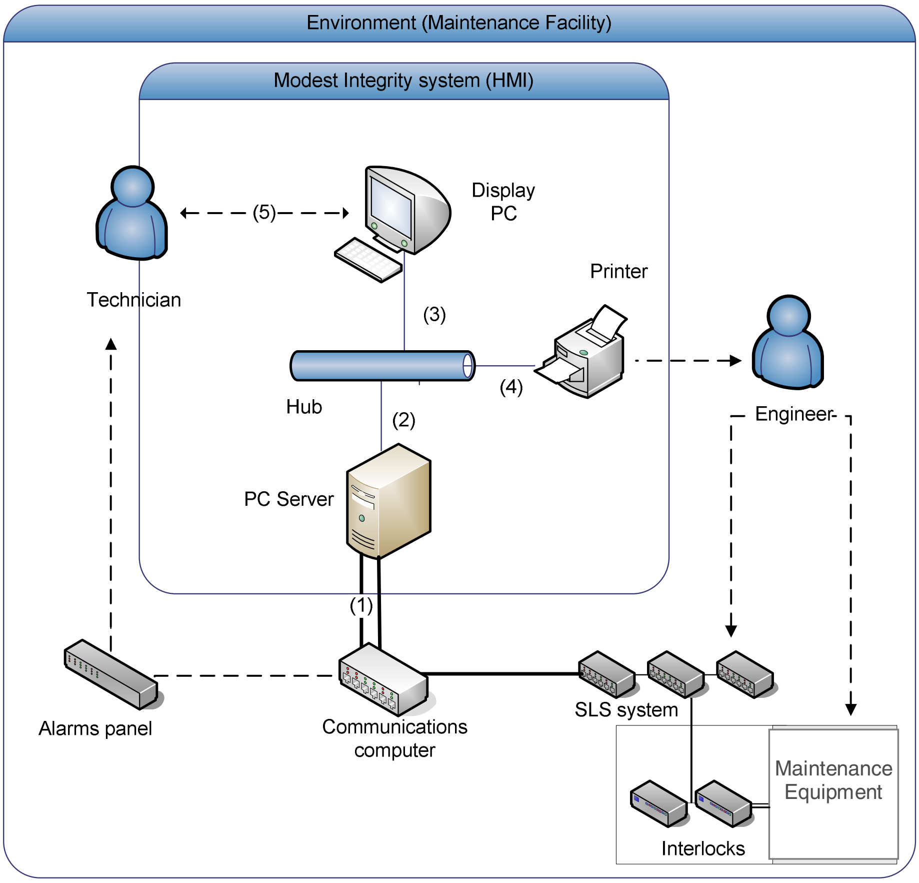

Fig. 3. HMI System Architecture and Environment

The HMI presents information on the plant status (SLS inputs) and state of the SLS outputs. This information is used by the plant operators (technicians) to diagnose plant set-up and configuration problems and also by engineers to identify UMF faults.

The example system and its context are shown in Fig. 3. Note that the links numbered 1 to 5 identify the PC interfaces that are a potential source of hazardous failures.

Operating Context. The system's immediate application environment consists of:

- a high-integrity logic system, known as the SLS, which performs safety interlocking for the UMF
- a communications computer for data acquisition from the SLS
- a hard-wired alarm facia panel driven directly by the SLS
- the UMF Technician, who is the first-line user of the HMI system to diagnose problems in the UMF plant interfaces
- the UMF Engineer, who is the person with the role to diagnose faults on the plant.

The system's broader engineering context is the rest of the UMF system which is responsible for assuring safety (via the use of interlocks) of the maintenance equipment. The HMI does not have any active safety role. However, the UMF technician, who acts partly on the information presented to him by the HMI system, can change the plant set-up and configuration as a result. His actions can therefore affect the state and serviceability of the SLS and of the UMF more generally.

The boundary of the HMI system therefore includes not only the technical elements above but also the technician who is shown straddling the system's boundary.

The Modest Integrity System. The main functions of the HMI are listed below:

- monitor and log time-stamped data from the SLS
- present plant status information, including external panel alarms, gathered by the SLS to diagnose the plant sensors
- present faults reported by the SLS itself, e.g. when a card within that system malfunctions
- produce hardcopy reports of the logged data to use in analysis by the maintenance engineer for confirming the correct operation of plant sensors

The HMI design comprises:

- A PC-based server, connected to the communications computer via dual-redundant RS422 links.
- A display terminal, also a PC, for displaying the data produced by the server to the UMF maintenance technician.
- A printer, for making hardcopies of the logged data.
- An Ethernet network and a hub.

The PC server uses the Microsoft Windows NT operating system. The display PC runs Windows CE 3.0. Application software running on the PC server and the client implement the monitoring and display functions.

NSPCs in the System. As can be seen in Fig. 3, the proposed design for the HMI system makes use of several NSPCs. These non-assured components are:

- the server PC (runs the main application software)
- the display PC (runs the main application software)
- the printer (contains firmware and is configurable to an extent)

It is clearly necessary for the assessment to consider the two PC components in detail, since they provide the essential functionality of the system.

5.2. Step 2: NSPC Feasibility Assessment

For the purposes of this illustration we will focus on one of the NSPCs (the PC server). This step assesses whether the NSPC is "fit for purpose" by considering whether:

- the functions performed are within the capability of the NSPC
- the performances demands are within the capability of the NSPC

This assessment can be supported by NSPC Annex guidance for a PC.

NSPC Functional Suitability. The overall functions provided by the PC components within the system are:

- data acquisition from the communications computer
- data display and logging (i.e. data monitoring)

There is Annex guidance on the types of function that can be performed by PCs. The results of the functionality assessment are shown in Table 1.

Table 1. Function assessment

Function type	Annex Guidance	Assessment
Monitoring interface	PC can be used to monitor non-critical data, i.e. system operation can continue without the displayed data.	Compliant The data collected and logged by the server PC is not essential for UMF operation.
Data acquisition, display, plotting or logging	PC can only be used to acquire and display / plot non-critical data, i.e. the system or environment can continue to operate without the acquired data.	Compliant The UMF can operate without the HMI display.

NSPC Performance Assessment. The performance demands are assessed using guide figures from the Annex as shown in Table 2.

Table 2. Performance assessment

Attribute	Target Level	Guide Limit	Assessment
Availability	> 95%	N/A (depends on MTBF)	Feasible. Based on guide figures for hardware and software MTBF and hardware repair and software recovery times.
Reliability	> 100 hours MTBF	1,000 hours for the Windows operating system 10,000 hours for the hardware	Feasible. The limiting factor in this case is the reliability of the application software.
Safe failure fraction	90%	90% if PC supervised *(50% if PC unsupervised)*	Feasible. Provided checks can be implemented that are 90% effective at alerting the operator to possible corruption in the displayed data.
Time response	< 5 seconds data request service time < 10 seconds data poll rate	1 second best case 10 seconds worst case	Feasible. Assuming lock-down of the PC server operating system.
Security	Medium	Medium	Feasible.
Usability	Medium	High	Feasible.

NSPC Assessment Conclusions. The overall conclusion is that a PC can be used as a server provided that:

- The operating system is locked-down
- The reliability of the application software meets the target.

5.3 Step 3: Failure Identification and Analysis

A failure analysis can be used to identify hazardous failures of the NSPCs. This could be done using a hazard analysis [1, 5] or failure modes and effects analysis [2]. The analysis must consider potential failures on all the relevant interfaces with the NSPCs (see the links numbered 1 to 5 in Fig. 3). The results of the Hazard analysis of the HMI system are shown in Table 3.

We could also consider any known vulnerabilities of the NSPC at this stage. These would be listed in the relevant NSPC Annex. However the “lock-down” guidance can

Table 3. Hazard Analysis Summary

Ref.	NSPC	Hazardous failure
HF1	Display PC, Server PC	Incorrect logic state data is displayed to the technician. • Wrong status data • Missing status data
HF2	Display PC, Server PC	Out-of-date logic state data is displayed to the technician.
HF3	Server PC	Misleading plant history is logged or reported. The history recorded in the logs, or recovered from them, may be misleading in the following ways: 1. Incomplete history because of missing data. 2. Incorrect history because of wrong data. 3. Incorrect history because of stale data.

be viewed as an alternative because a standard set of countermeasures are identified to address vulnerabilities and so enhance the integrity and security of the component.

5.4 Step 4: Safety Impact Assessment

In this step we consider whether the existing mitigations to the hazardous NSPC failures are adequate. The current mitigations are listed in Table 4.

Table 4. Existing Mitigations

Ref	Description of mitigation
M1	(External to the HMI system) The SLS safety hardware has fail/OK status indicators on each card (reducing the risk of the maintenance technician changing the wrong card).
M2	(External to the HMI system) The equipment is tested after repair – if the wrong board is replaced the HMI would still give the same (misleading) state after repair.
M3	(External to the HMI system) There is a separate alarm panel (driven by the safety SLS equipment) so the control engineer is not relying on the HMI being the primary information source.
M4	(Internal) Display PC detects failure of server to respond within a set timeout period and the display is “greyed-out” to indicate loss of communications with server.

These defences are assessed for adequacy against the identified hazardous failures shown in Table 5.

As the defences are only considered to be partially effective, we need to consider additional mitigation options, specifically enhancements to:

- Assist the detection of data corruption or incorrect processing by the software in the display PC or server PC, or require specific activities to be carried out by the Supplier to demonstrate that the software has the requisite integrity.
- Improve the detection of software hangs.

The options for additional mitigations are addressed in Step 5.

Table 5. Adequacy of Existing Mitigations

Ref	NSPC	Failure	Mitigations	Adequate?
HF1	Display PC, Sever PC	Incorrect display.	M1: SLS show plant status M2: Test after repair	Yes.
HF2	Display PC, Server PC	Out-of-date display	M4: Timeout on loss of communications	Partially. Not effective if the display PC software itself hangs.
HF3	Server PC	Misleading plant history	M3: Alarm data display on separate SLS panel	Partially. Only provides a snapshot of the current status of the plant. The diagnosis requires an accurate history.

5.5 Step 5: Identify Additional Mitigations

The guidance has a checklist of possible mitigations as shown in Table 6. Their applicability has to be interpreted in the context of the particular system design.

Table 6. Mitigation Options

Technical	Procedural
Partitioning	Periodic proof tests
End-to-end integrity checks	Regression tests
External safety checks/interlocks	Fault reporting procedures
Watchdogs	Change control procedures
Clear user interface	Security and access control procedures
Status indication	Procedures for operating under failure conditions
Redundancy	
Diversity	Staff competence and training

For retrospective system assessment, any additional barriers are likely to be procedural. For new systems, there is more scope for incorporating additional technical barriers at an early stage in the design. For brevity we will focus on the applicability of the potential technical barriers to the HMI system. The results are summarised in Table 7.

Table 7. Review of the Applicability of Technical Options to the HMI

Option	Assessment
Partitioning	N/A The system's environment is already well partitioned.
End-to-end integrity checks	Feasible. Dummy input signals can be added for checking the correctness of the entire data processing done by the HMI system. Each of these dummy signals displays a set known pattern that can be searched for in the logs and observed on the display.
External safety checks/interlocks	N/A The PCs do not perform a control function.
Watchdogs	Uncertain. More applicable to control computers.
Clear user interface	Already covered. HMI display conforms to human factors guidelines.
Status indication	Feasible. A heartbeat indicator should be displayed on the display PC to show that the screen is being updated and that the data acquisition part of the system is active.
Redundancy (communications)	Not required. Dual communications links are already provided between the HMI and the communications computer.
Diversity	Not practicable for PC based systems.

Based on this assessment, the design is modified to include two additional mitigations (see Table 8):

Table 8. Design Enhancements

Ref.	Recommended additional mitigation	Mitigates
M5	End-to-end checks using additional dummy UMF plant input signals. Reveals failures in data acquisition, processing and display / logging functions of the server PC and display PC.	HF1 (wrong data) HF3 (wrong logs)
M6	Status indication. Addition of a heartbeat or liveness indicator, which would enable the operator to detect whether the display screen is being updated.	HF2 (stale data)

5.6 Step 6: Collect and Communicate Evidence Requirements

In step 6 the evidence requirements from the previous steps are collated and assigned for implementation (typically to the system supplier or the plant operator). In Step 2, a

Table 9. Additional Requirements

Source	Requirement	Performed by
Step 2	Lock down the operating system in the server PC	Supplier
Step 2	Demonstrate reliability of the application software	Supplier
Step 2	Demonstrate application conforms to standards	Supplier
Step 5	M5: End-to-end integrity check. Implement dummy plant input scan software change	Supplier
Step 5	M5: Install dummy plant signals in UMF	Plant operator
Step 5	M5: Test of end-to-end check	Operator + supplier
Step 5	M6: Add status indication "heartbeat" software	Supplier
Step 5	M6: Evidence of correct operation	Operator + supplier

need was identified to demonstrate adequate reliability of the application software. This could be covered by compliance to appropriate standards (e.g., [3, 4]) and comprehensive functional testing. Step 5 identified additional mitigations that need to be implemented. We require evidence that these mitigations have been correctly implemented. The additional requirements are shown in Table 9.

5.7 Step 7: Evaluate the Evidence and Produce a Report

Once the changes have been implemented, the evidence is evaluated to assess whether it satisfactorily meets the evidence requirements. Based on the evidence produced and the analyses performed in the previous steps, an assessment report is produced, containing a clear recommendation as to whether the system can be accepted or not.

6 Concluding Remarks

The LowSIL approach has been developed and updated over a number of years and has been applied to a range of control and instrumentation systems used in nuclear power plants as a means of demonstrating adequate safety assurance when non safety-assured programmable components are used. Future developments are under consideration, primarily in the development of new Annexes for different operating systems, middleware and devices.

Acknowledgments. The authors wish to acknowledge the support of the CINIF research programme that funded the research presented in this paper.

References

1. IEC 61882: Hazard and operability studies (HAZOP studies) – Application guide (2001)
2. IEC 60812: Analysis Techniques for System Reliability – Procedure for Failure Mode Effects Analysis (1985)

3. IEC 61508-3: Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements, 1st edn. (1998)
4. IEC 62138: Nuclear power plants – Instrumentation and control important for safety – Software aspects for computer-based systems performing category B or C functions, 1st edn. (2004)
5. Kletz, T.A.: HAZOP and HAZAN, Identifying and Assessing Process Industry Hazards, 4th edn. Institution of Chemical Engineers (2006)